Amendments to the Specification:

(1) Please replace the paragraph numbered 0236 beginning at page 66, line 14, with the following rewritten paragraph:

[0236] — The following procedure can be used to compute the worst-case response time of each A-h-k-a process:

i:= 0;

failure= false;

while i ≤ number-of-A-h-k-a-processes and not(failure) do

begin

    if a_i ∈ A-h-k-a

    then

    begin

        $RE_{new_i} := c_{a_i}$;

        responsetimefound:= false;

        while not(responsetimefound) and not(failure) do

        begin

            $RE_{previous_i} := RE_{new_i}$;

            $RE_{new_i} := \underline{c_{a_i}} + \text{DelayA}(a_i, RE_{previous_i}) + \text{DelayP}(a_i, RE_{previous_i})$
$$+ \text{B}(a_i) + \text{GT}(a_i, RE_{previous_i});$$

            if $RE_{previous_i} = RE_{new_i}$

            then

            begin

                $RE_{a_i} := RE_{new_i}$;

                responsetimefound:= true;

            end

            if $(RE_{new_i} > L_{a_i})$

            then failure:= true;

        end;

    end;

    i:= i + 1;

end —

2

(2) Please replace the paragraph numbered 0372 beginning at page 122, line 14, with the following rewritten paragraph:

[0372]   — The following procedure can be used to compute the worst-case response time of each A-s-k process:

i:= 0;

failure= false;

while i ≤ number-of-A-s-k-processes and not(failure) do

begin

      if $a_i \in$ A-s-k

      then

      begin

          $RE_{new_i} := c_{a_i}$;

          responsetimefound:= false;

          while not(responsetimefound) and not(failure) do

          begin

              $RE_{previous_i} := RE_{new_i}$;

              $RE_{new_i} := \underline{c_{a_i}} + \text{DelayA}(a_i, RE_{previous_i}) + \text{DelayP}(a_i, RE_{previous_i}) + B(a_i)$;

              if $RE_{previous_i} = RE_{new_i}$

              then

              begin

                  $RE_{a_i} := RE_{new_i}$;

                  responsetimefound:= true;

              end

              if $(RE_{new_i} > \text{responsetimelimit})$

              then failure:= true;

          end;

      end;

      i:= i + 1;

end —